Developing New Processes for COTS-Based Systems

Although commercial off-the-shelf (COTS) products are becoming increasingly popular, little information is available on how they affect existing software development processes or what new processes are needed. At the Carnegie Mellon University's Software Engineering Institute (SEI), we are developing a process framework for working with COTS-based systems.

Lisa Brownsword, Tricia Oberndorf, and Carol A. Sledge
Software Engineering Institute

In response to the particular system circumstances their projects face, developers use various processes to create and maintain their custom-developed systems. Projects to create COTS-based systems (CBSs) thus run into difficulties in trying to follow custom-development processes.[1–3] The use of COTS products introduces new system circumstances, which then requires new software development processes. Although researchers and practitioners have been grappling with these new processes,[4,5] no one has yet comprehensively described the processes—engineering, business, and management—that arise or change in response to this growing use of COTS products and technology.

While such factors as domain and life cycle stage prevent any one process from working for all CBS projects, we can define and organize the activities common to most endeavors. At the Software Engineering Institute, we are developing a CBS process framework for guiding projects during their detailed planning.

The preliminary results described here arise from our work with over 30 medium-sized and large projects, ranging from business information management systems to embedded weapon and military command and control systems.[6–8] We have captured information about the engineering, business, and management practices that worked and those that did not. After analyzing this information to understand and characterize the common points of success and failure, we also identified where projects departed from traditional development processes. This article will identify the resulting process changes required to address these real-life lessons and articulate a framework for organizing the new and changed process elements.

CBS Process Drivers

In lieu of coding components, many CBS developers assume they can just "plug in" COTS products. They assume using COTS products will shorten their programming and testing effort, with little other lifecycle process effect. Our experiences at the SEI, however, invariably show that the use of COTS products has more pervasive ramifications.

We define a COTS product as one that is

0740-7459/00/$10.00 © 2000 IEEE

- sold, leased, or licensed to the general public;
- offered by a vendor trying to profit from it;
- supported and evolved by the vendor, who retains the intellectual property rights;
- available in multiple, identical copies; and
- used without source code modification.

New process drivers flow both from this COTS product definition and from the consequences of assembling systems from COTS products. These new process drivers are:

- CBS development is an act of composition.
- The realities of the COTS marketplace shape CBS development.
- CBS development occurs through simultaneous definition and trade-off of the COTS marketplace, system architecture, and system requirements.

CBS Development Is an Act of Composition

COTS-based system development involves composition and reconciliation, whereas custom system development is an act of creation. Custom development starts with the system requirements and creates a system that meets them; the engineers are producers. However, COTS-based system development starts with a general set of requirements and then explores the marketplace's offerings to see how closely they match the needs; the engineers are consumers, who then integrate the products they buy into a system that meets the need. The nature, timing, and order of activities performed and the processes used differ accordingly.

The Realities of the COTS Marketplace Shape CBS Development

The marketplace affects the nature and evolution of a COTS-based system. Inherent marketplace characteristics help determine the future of a COTS-based system endeavor:

- There is frequent, continuous change in COTS products and the marketplace.
- The marketplace, not the needs of any particular system, drives COTS products.
- Products have built-in assumptions about how they will be used; these might not match the system users' processes, resulting in clashes.
- Licensing and data rights will affect cost, architecture, and user processes.
- Projects have limited control over a COTS product's release frequency or content.
- Projects have limited visibility into COTS products' source code and behavior.
- Products are built on architectural assumptions that can vary across system components and could conflict with an evolving system architecture.
- COTS products will have interdependencies.

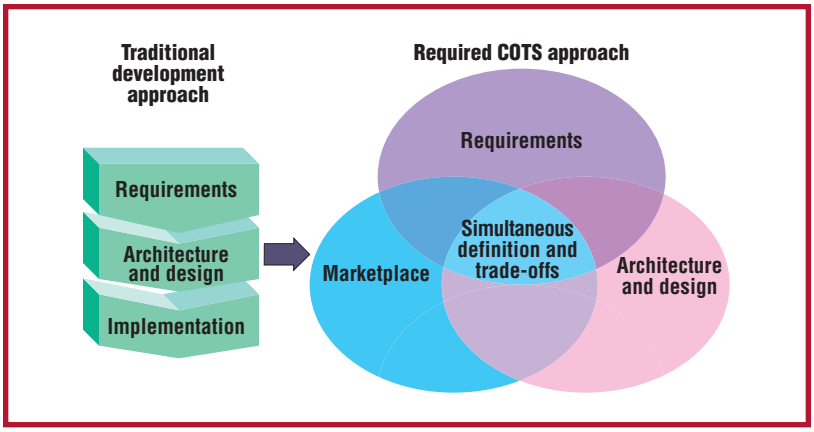

Figure 1. Traditional versus COTS-based approaches.

CBS Development Occurs through Simultaneous Definition and Trade-Offs

The third CBS process driver is really a consequence of the first two: the approach to system development for COTS-based systems requires a fundamental change, as pictured in Figure 1. On the figure's left is a traditional custom-development approach in which the development team identifies requirements, defines an architecture, and then undertakes (custom) implementation. (We use the term *requirements* here in the broadest sense. Requirements include nonfunctional requirements, end-user processes, and other constraints such as cost and schedule, not just functional requirements.)

If we applied this approach to COTS-based systems, the marketplace would not likely yield any products that fit the a priori requirements and architecture. Instead, with COTS-based systems, system developers must consider requirements, architecture, and marketplace *simultaneously*, as pictured on the right of Figure 1. Any of the three might affect the other two, so none

can proceed without knowledge and accommodation of the others. Furthermore, the activities performed for CBSs are cyclic: these trade-offs recur frequently throughout the system's lifetime.

This fundamental change necessitates differences in the processes used to develop and maintain systems with COTS products and technologies. When processes change, people must change as well. In other words, the move to COTS-based systems development is not just an engineering or technical change—it is a business, organizational, and cultural change, too.

CBS Process Framework

To understand the process changes generated by the use of COTS products, we identified the activities that are either new for COTS-based systems or were present in custom development but change for CBS development. These activities fall into four major *activity areas*: engineering, business, project-wide, and contract. (Some might call these business processes, but we use the term activity area because it focuses on differences in activities, and the work has not yet progressed to the articulation of full processes.) The engineering, business, and contract categories are straightforward. The project-wide category accounts for activities that are not contained in one area but span multiple areas. These four activity areas constitute the top-level structure of the CBS process framework.

Within each of these activity areas, we categorize the new and changed activities into *activity sets*. Each activity set operates continuously. There is no implied sequence in an activity area. For each activity set, we identified its scope, the key differences from custom development, the activities themselves, and usage guidance or tips to consider in defining and applying the activities.

Sometimes the differences between CBS and custom-development processes are subtle. Often the differences are not in what is done but rather how or when or with what marketplace considerations the CBS activity occurs. For example, the steps in CBS risk management are the same steps used in any risk-management effort; the difference derives from the nature of COTS risks and the mitigations required. Similarly, it is not unusual in custom development to make trade-offs between requirements and architecture, but the marketplace considerations change the balance and nature of some such trade-offs.

Because this work emphasizes the new and changed activities, the activity areas do not cover everything that is done on a successful project. This work builds on good basic engineering and management practices that have been widely adopted in the software engineering community over the past decade. (The SEI has produced other process frameworks, known as *Capability Maturity Models*. CMMs focus on custom-developed, not COTS-based, systems and so do not address the issues of concern here. This process framework is an articulation of COTS-related activities, not an attempt at a maturity model.)

Figure 2 shows the parts of the CBS process framework that we discuss in this article. (In the interest of space, we have chosen not to discuss the contract activity area here.)

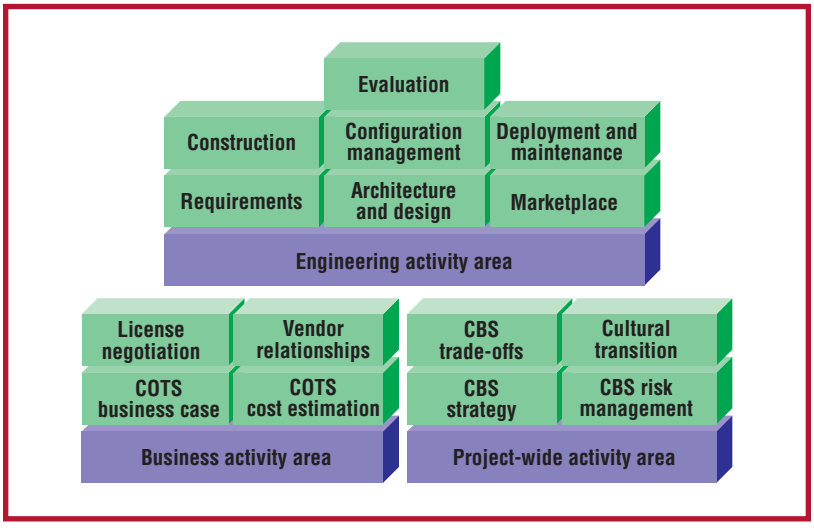

Figure 2. COTS-based systems activity areas.

Engineering Activity Area

The engineering activity area is associated with a system's technical conceptualization, construction, and maintenance. These activities implement the approach to CBS development in Figure 1. In particular, the requirements, marketplace, and architecture and design activity sets must operate concurrently and in cooperation with one another. See the "Engineering Activity Area" box.

Requirements activity set. This set defines, prioritizes, and consstrains the CBS to be

Engineering Activity Area

Requirements activities
- Determine and prioritize the requirements' negotiable and nonnegotiable elements.
- Understand the essential elements of the end-user processes before committing to the marketplace.
- Modify end-user processes as necessary to resolve process and product mismatch.
- Negotiate requirements changes as part of COTS-based trade-offs.
- Dynamically reflect results of COTS-based trade-offs in the requirements description.
- Periodically reexamine COTS products for opportunities to optimize user processes.

Architecture and design activities
- Generate alternative architectures and designs and select candidate COTS products and technologies using COTS evaluation results.
- Validate architecture early (using an executable architecture approach if possible).
- Understand and reflect marketplace impacts in the architecture and design.
- Reflect results of COTS-based trade-offs in the architecture as they occur.

Marketplace activities
- Create and maintain relevant and *current* knowledge of the available and emerging marketplace through market research and technology watch and participation in industry and user groups.
- Augment marketplace knowledge using results of trade-offs, prototypes, and pilots: "Try before you buy."
- Alert project staff of promising new technologies and products.

Construction activities
- Create "glue" to provide any necessary adaptation of COTS products into the system.
- Perform any tailoring (but not product modification) required to use a COTS product in the system.
- Integrate and test early and continuously to discover product clashes as quickly as possible.
- Characterize COTS products continuously.
- Continuously determine impact of product upgrades.

Configuration management activities
- Identify configuration baselines.
- Receive and process COTS upgrades, patches, bug fixes, and so forth.
- Systematically control changes to configurations.
- Release new system versions.
- Coordinate with construction and license negotiation activity sets.

Deployment and maintenance activities
- Plan the support to accommodate COTS realities.
- Incorporate new product releases, using construction activities.
- Re-tailor COTS products.
- Create and update documentation and training for product upgrades.
- Define and provide end-user support for COTS products.
- Engineer (including reintegrate) and coordinate new product releases from multiple vendors into system release.
- Manage licenses.

Evaluation activities
- Plan the evaluation.
- Design the evaluation (turn requirements and architectural attributes into evaluation criteria; choose a technique for aggregating scores; choose an assessment approach).
- Locate potentially relevant technologies and products.
- Perform analyses (initial and detailed).
- Document and share information for decision-making.

fielded, accounting for functional and nonfunctional requirements, end-user processes, business drivers, the operational environment, and constraints (such as policies, schedules, and budgets). Mismatches between end-users' processes and the processes embodied in COTS products will occur. These differences will constrain both the system requirements and the project's ability to leverage the marketplace. Such mismatches demand a different approach to requirements development, emphasizing a new source of inputs (the marketplace) and a new willingness of stakeholders to negotiate requirements. Early and continual stakeholder involvement is needed for deciding the potential compromises between requirements and available COTS products and technologies.

Marketplace activity set. This set bounds the COTS marketplace elements that are relevant to the system over its lifetime. The activity set governs the conduct and documentation of market research and participation in negotiations and trade-offs that the COTS technologies and products marketplace affect. These activities are largely new to the system development and maintenance process. COTS-based systems are evolution-

Business Activity Area

COTS business case activities

- Determine CBS success factors.
- Conduct preliminary study into the feasibility of a COTS-based solution.
- Identify key COTS-related assumptions.
- Articulate alternatives to be analyzed.
- Analyze CBS financial implications.
- Analyze (COTS and non-COTS) alternatives and determine recommendations.
- Revisit the COTS business case periodically and at key events.

COTS cost estimation activities

- Identify cost factors, including those new or impacted by the use of COTS products and services.
- Select and calibrate COTS cost estimation model and techniques.
- Estimate costs.
- Provide cost estimates to other activity sets.
- Track CBS actuals versus estimates.
- Maintain COTS cost estimation models and techniques based on collected data and marketplace trends.

Vendor and supplier relationships activities

- Develop strategy to create and manage vendor and other supplier relationships.
- Engage in meetings and exchanges with vendors, suppliers, and related groups.
- Understand and monitor vendor's and supplier's long-term approach and plans for maintenance.
- Establish liaisons with vendor's and supplier's other customers.
- Encourage and facilitate working relationships among vendors and suppliers.

License negotiation activities

- Investigate licensing alternatives and costs; capitalize on enterprise licensing opportunities.
- Incorporate nonstandard provisions.
- Negotiate licenses.

ary—subject to the evolutionary demands created by the marketplace in addition to the usual changes in end-user needs. Marketplace-driven changes result from both the natural ebb and flow of products and technologies and the discovery of new capabilities in the marketplace.

Architecture and design activity set. This set captures decisions about the structure of the components, interfaces, and relationships of a system's components, and the principles and guidelines governing their design and evolution over time. The marketplace's evolutionary nature strongly affects the COTS-based system architecture and design, which must now withstand years if not decades of change. An architecture that allows efficient evolution of a system is a strategic asset for COTS-based systems—it is the only thing the project owns. Development of such an architecture must occur in concert with the evolving requirements and product decisions, creating trade-off situations custom development will rarely encounter.

Construction activity set. This set addresses implementation of custom components, COTS integration, and system integration and test. You compose and integrate a COTS-based system from available parts. (Construction includes more than programming. Programming is often far less prominent for COTS-based systems than for custom-developed ones.) This involves a profoundly different mindset from that of custom development and requires a significantly different set of skills. COTS product modification is often a temptation; avoid it whenever possible, as "modified COTS" is an oxymoron. A project must consider system lifetime costs of COTS product tailoring or modification as part of architectural and product selection decision-making. Testing does not go away. Rather, its nature shifts from white box (using knowledge of the source code and design) to black box (without knowledge of the source code or design), and system-level testing receives increased attention.

Configuration management activity set. This set establishes and maintains system artifact integrity and traceability throughout the CBS's lifetime. It will start earlier in the process, from the first evaluation of candidate products. The managed baseline will change more frequently due to changes in COTS products and the marketplace. The CM system will have to track new artifacts, including COTS product versions, tailorings, patches, installation procedures, and possibly license management information.

Deployment and maintenance activity set. This set encompasses initial and continuing delivery of a COTS-based system to end users and system maintenance (routine and enhancements). The traditional separation of development and maintenance blur and become indistinguishable with COTS-based sys-

tems. Maintenance events, such as product upgrades, will occur before the system's initial delivery. Construction activities such as product selection, test, and integration will be necessary during maintenance. A major challenge during deployment and maintenance is striking a balance between system stability and the need to stay current with the marketplace.

Evaluation activity set. This set examines COTS products and technologies to gather information in support of making CBS decisions throughout the system lifetime. It is a new part of the process, not found in custom development. COTS evaluation takes different forms under different circumstances, such as market research for initial product screening or gap analysis for more detailed understanding of functional capabilities and process mismatch. COTS evaluation begins with the first idea for a system, underlying all the other activities continuously throughout the CBS lifetime. This suggests dedicated evaluation resources in the form of people, software, hardware, and facilities, as market information's usual half-life is very short—about six months.

Business Activity Area

The business activity area helps in developing the business case and cost estimates, negotiating licenses, and managing supplier relationships. COTS product and technology decisions are not just engineering decisions; they are also business decisions. Many activities (see the "Business Activity Area" box) in the business activity area require information from the engineering activity area and vice versa. For example, creating a CBS business case requires detailed COTS product information derived from the marketplace and evaluation activity sets and architectural and design prototypes from the architecture activity set. Furthermore, the licensing arrangements available for a key COTS product might be cost-prohibitive or incompatible with the selected system architecture.

COTS business case activity set. This set provides the basis for make-versus-buy decisions for an entire system or an individual component. It covers the information gathering and analyses necessary to reach a recommendation regarding which of several alternative COTS or custom solutions to choose, using many of the other engineering and business activity sets. COTS business cases must incorporate the total cost of COTS product ownership across the system life, not just initial product purchase costs. A COTS business case takes into account market research and trend analysis, gap analysis, investigations of vendor health and practices, and detailed product usage through prototypes, demonstrations, and pilots.

COTS cost estimation activity set. This set involves the identification of new and changed costs associated with the incorporation of COTS products in a system and their inclusion in a cost estimation model or technique. New COTS-related cost factors might include reacting to new product releases and marketplace changes (including end-of-life events), technology forecasting, market research, engineering an evolvable architecture, licensing, integration, and reintegration. CBS cost estimation must account for all the differences implied by the CBS process drivers. COTS cost estimation techniques and models are in their infancy.

Vendor and other supplier relationships activity set. This set determines candidate vendors and other suppliers (such as corporate sister organizations or another government agency or department) and the nature of relationships with them. It is built on cooperative exchanges between the acquirer and the vendor or supplier that explore current and future vendor and supplier plans as well as project plans, provide insights into product releases, and represent a means to influence vendor and supplier plans. Developing and maintaining effective relationships with vendors and suppliers is a new activity for many projects. *Vendor* is not a new term for *contractor*: contractors are directed to perform agreed upon work within cost, schedule, and quality parameters, but vendors and other suppliers do not work that way.

License negotiation activity set. This set examines what the vendor offers with respect to terms, conditions, and costs for a given product for use by an organization over a particular period of time. Based on the organization's situation and needs, this set of

An architecture that allows efficient evolution of a system is a strategic asset for COTS-based systems—it is the only thing the project owns.

Project-Wide Activity Area

CBS strategy activities

- Identify CBS goals, constraints, and assumptions.
- Coordinate activities for all other activity sets into corresponding CBS plan, including contingency plans.
- Reassess CBS strategy as necessary.

COTS risk-management activities

- Identify and prioritize COTS risks.
- Analyze COTS risks.
- Plan and institute COTS risk mitigations.
- Track COTS risks and effectiveness of COTS risk mitigation.
- Revisit and revise COTS risk management as necessary.

CBS trade-offs activities

- Determine affected stakeholders and involve them.
- Identify where CBS trade-offs are needed.
- Gather sufficient information to make informed CBS trade-offs.
- Select a resolution.

Cultural transition activities

- Assess readiness of all stakeholders for transition to CBS.
- Identify CBS skill sets required.
- Secure buy-in of senior leadership and project staff.
- Develop and implement a strategy for accomplishing the transition.
- Collect and share CBS information (such as market research, technology trends, guidelines, exemplary architectures, strategies, licensing agreements, lessons learned, and decision rationale).
- Encourage CBS champions.
- Provide incentives.
- Train everyone.

activities negotiates the licenses that best suit both parties. License agreements embody the project's vendor relationships, including integration support. They can also address the vendor's commitment to include in future product releases any project-specific product modifications. These agreements must withstand changes over time, such as product splits and license transfers. Typically the vendor has a set of licenses they offer, but other options may exist.

Project-Wide Activity Area

The project-wide activity area (see the "Project-Wide Activity Area" box) spans and unites the engineering and business activity areas in a COTS-based system's development and maintenance.

CBS strategy activity set. This set derives an approach to COTS-based system development that will meet CBS objectives within project constraints over the system's life. The CBS strategy sets the stage for how a project will conduct all other activities. For example, the CBS strategy governs to what depth a COTS business case will be done, what investment to make in vendor relationships, and what development approach will best support a CBS project. A project will need to reevaluate its CBS strategy periodically and adjust its plans and actions accordingly; marketplace volatility will force reevaluation more frequently than for custom-developed systems.

CBS risk-management activity set. This set identifies COTS risks as early as possible, adjusts the strategies and plans to manage those risks, and develops and implements a COTS risk-management process as an integral part of an organization's overall CBS development. The risk-management process does not differ significantly for CBS, but the risks do. Given marketplace volatility, COTS risks tend to change more rapidly than with custom systems. Examples of common COTS risks include a key vendor going out of business or an engineer's inability to integrate two selected products.

CBS trade-off activity set. This set seeks to identify and balance the contentions that arise among two or more CBS activity sets in the engineering, business, and project-wide activity areas. It ensures that trade-offs occur at the appropriate time in the appropriate context and with the appropriate rationale. Engineering has always dealt with trade-offs. With COTS-based systems, new trade-off considerations arise, such as requirements the products do not meet; effects of licenses on design decisions; a vendor's or a supplier's market share; architectural mismatch among COTS products; long-term viability of a technology, product, or vendor; and the match or mismatch of COTS product processes and existing end-user processes. To compound the trade-off issues for a COTS-based system, a project does not control many of those new sources of contention. Attempting to compensate by modifying COTS products generally is not a viable option. Even if the project can get the source code, COTS modification relies on understanding unfamiliar code and prevents cost-effective upgrades.

Cultural transition activity set. This set tries to incorporate the new CBS mindset and development and maintenance practices into people's daily activities. It seeks to manage individual and organizational changes critical to achieving strategic CBS business objectives. COTS-based systems represent a change for everyone in an organization—not just technical personnel—which requires new roles and skills. Failing to pay attention to the cultural-transition issues could result in an insurmountable barrier to CBS success. When the pace of change accelerates, flexibility is a business imperative: a project does not have the time to recover from problems it could have avoided. Information sharing can help save others from repeating known mistakes.

Our CBS process framework is preliminary. To date, no one project has pursued their work according to this set of ideas. Any framework requires a great deal of application to vet and tune. Projects can use this framework and its contents in several ways: to determine what processes are required to effectively leverage the COTS marketplace, to identify the difference between existing processes and those required, or to determine a suitable migration path.

We have identified two kinds of validation activities. One involves using applicable activity sets on projects. We plan to do this with our customers. We would invite any reader who chooses to work with some or all of these activities to share their results with us and thus contribute to these activities' improvement as a community resource. The second kind of validation activity involves further study of the processes that experienced CBS practitioners use through interviews, by which we learn how they have structured their processes. We anticipate using the results from both kinds of validation activities to evolve the CBS process framework that is described here.

References

1. B. Boehm and C. Abts, "COTS Integration: Plug and Pray?" *Computer*, Vol. 32, No. 1, Jan. 1999, pp. 135–138.
2. L. Brownsword, D. Carney, and P. Oberndorf, "The Opportunities and Complexities of Applying Commercial-off-the-Shelf Components," CrossTalk: *The J. of Defense Software Eng.*, Vol. 11, No. 4, Apr. 1998, pp. 4–6.
3. D. Garlan, R. Allen, and J. Ockerbloom, "Architecture Mismatch: Or Why It's Hard to Build Systems Out of Existing Parts," *Proc. Int'l Conf. Software Eng.*, IEEE Computer Soc. Press, Los Alamitos, Calif., 1995, pp. 179–185.
4. G. Fox, S. Marcom, and K. Lantner, "A Software Development Process for COTS-Based Information System Infrastructure," *CrossTalk: J. Defense Software Eng.*, Vol. 11, Nos. 3–4, Mar. and Apr. 1998, pp. 20–25 and 11–13, respectively.
5. D. Reifer, *Product Line Management: Best Acquisition Processes/Practices*, Southern Calif. Software Process Improvement Network (SPIN), Univ. of California at Irvine, Irvine, Calif., 1999.
6. P. Oberndorf, L. Brownsword, and C. Sledge, *An Activity Framework for COTS-Based Systems*, SEI Technical Report CMU/SEI-20000-TR-101, SEI, Carnegie Mellon Univ., Pittsburgh, 2000.
7. C. Sledge and D. Carney, *Case Study: Evaluating COTS Products for DoD Information Systems,* CBS monograph series, SEI, Carnegie Mellon Univ., Pittsburgh, Penn.; www.sei.cmu.edu/cbs/monographs.html, current June 1998.
8. S. Hissam and D. Plakosh, *COTS in the Real World: A Case Study in Risk Discovery and Repair*, SEI Tech. Note CMU/SEI-99-TN-003, SEI, Carnegie Mellon Univ., Pittsburgh, June 1999.

About the Authors

Lisa Brownsword is a senior member of the technical staff in the CBS Initiative at the SEI. Before joining the SEI, she was employed at Rational Software Corporation providing consulting to managers and technical practitioners in the use of and transition to software engineering practices, including CASE, architecture-centered development, product lines, and object technology. She received a BA in computer science from San Diego State University. She is a member of the ACM and an affiliate member of the IEEE Computer Society. Contact her at the SEI, Carnegie Mellon Univ., 4301 Wilson Blvd., Suite 902, Arlington, VA 22203-4191; llb@sei.cmu.edu.

Tricia Oberndorf is a senior member of the technical staff at the SEI. She is a part of the CBS Initiative and concentrates on the investigation of acquisition, management, and open system issues. Prior to coming to the SEI, she was with the Navy for more than 19 years, working on CASE environments. She received a BS in mathematics (computer science option) from Oregon State University and an MS in computer science from the University of California at San Diego. She is a member of the ACM. Contact her at the SEI, Carnegie Mellon Univ., 4500 Fifth Ave., Pittsburgh, PA 15213-3890; po@sei.cmu.edu.

Carol A. Sledge is a senior member of the technical staff at the SEI. She is a member of the CBS Initiative at the SEI and concentrates on open systems and acquisition and management issues of COTS-based systems. She has 22 years of experience, primarily acquiring, developing, and supporting large, multiplatform product line systems. She received a BS in mathematics, an MS in computer science, and a PhD in computer science from the University of Pittsburgh. She is a member of the ACM and the American Management Association and an affiliate member of the IEEE Computer Society. Contact her at the SEI, Carnegie Mellon Univ., 4500 Fifth Ave., Pittsburgh, PA 15213-3890; cas@sei.cmu.edu.